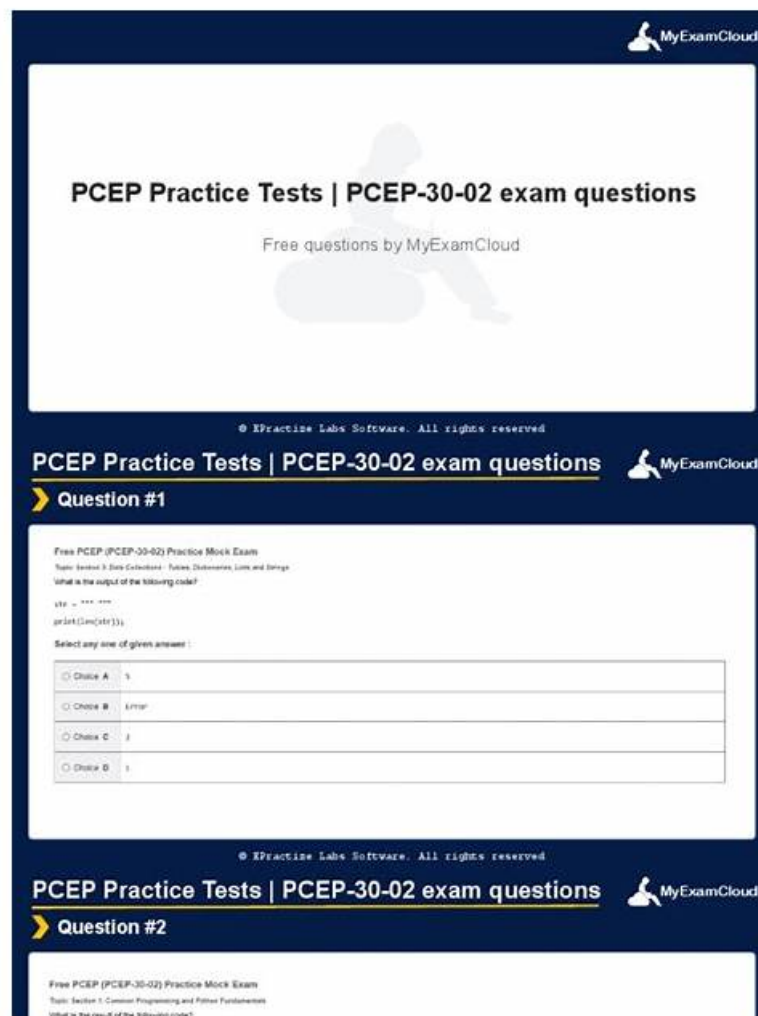


# PCEP-30-02 Praxisprüfung & PCEP-30-02 Lernhilfe



2025 Die neuesten ZertFragen PCEP-30-02 PDF-Versionen Prüfungsfragen und PCEP-30-02 Fragen und Antworten sind kostenlos verfügbar: <https://drive.google.com/open?id=1-heru-TM77tTSUOJ89PsX1UjLThaiK8>

Wenn Sie die schwierige Python Institute PCEP-30-02 Zertifizierungsprüfung bestehen wollen, ist es unmöglich für Sie bei der Vorbereitung keine richtige Schulungsunterlagen benutzen. Wenn Sie die ausgezeichnete Lernhilfe finden wollen, sollen Sie an ZertFragen diese Prüfungsunterlagen suchen. Wir ZertFragen haben sehr guten Ruf und haben viele ausgezeichnete Dumps zur Python Institute PCEP-30-02 Prüfung. Und wir bieten kostenlose Demo aller verschiedenen Dumps. Wenn Sie suchen, ob ZertFragen Dumps für Sie geeignet sind, können Sie zuerst die Demo herunterladen und probieren.

Mit den Schulungsunterlagen zur Python Institute PCEP-30-02 Zertifizierungsprüfung von ZertFragen können Sie die neuesten Fragen und Antworten zur Python Institute PCEP-30-02 Zertifizierungsprüfung bekommen und somit die Python Institute PCEP-30-02 Zertifizierungsprüfung erfolgreich einmalig bestehen. Die Python Institute PCEP-30-02 Zertifizierungsprüfung ist nützlich für Ihre Berufskarriere. Die Schulungsunterlagen zur Python Institute PCEP-30-02 Zertifizierungsprüfung von ZertFragen garantieren, dass Sie die Fragen sowie deren Konzept verstehen können.

>> PCEP-30-02 Praxisprüfung <<

## PCEP-30-02 Lernhilfe - PCEP-30-02 Examengine

Hohe Effizienz ist genau das, was unsere Gesellschaft von uns fordern. Die in der IT-Branche arbeitende Leute haben bestimmt das erfahren. Möchten Sie so schnell wie möglich die Zertifikat der Python Institute PCEP-30-02 erwerben? Insofern Sie uns finden, finden Sie doch die Methode, mit der Sie effektiv die Python Institute PCEP-30-02 Prüfung bestehen können. Die Technik-Gruppe von uns ZertFragen haben seit einigen Jahren große Menge von Prüfungsunterlagen der Python Institute PCEP-30-02 Prüfung

systematisch gesammelt und analysiert. Außerdem haben Sie insgesamt 3 Versionen hergestellt. Damit können Sie sich irgendwo und irgendwie auf Python Institute PCEP-30-02 mit hoher Effizienz vorbereiten.

## Python Institute PCEP-30-02 Prüfungsplan:

| Thema   | Einzelheiten                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Thema 1 | <ul style="list-style-type: none"><li>• Computer Programming Fundamentals: This section of the exam covers fundamental concepts such as interpreters, compilers, syntax, and semantics. It covers Python basics: keywords, instructions, indentation, comments in addition to Booleans, integers, floats, strings, and Variables, and naming conventions. Finally, it covers arithmetic, string, assignment, bitwise, Boolean, relational, and Input</li><li>• output operations.</li></ul> |
| Thema 2 | <ul style="list-style-type: none"><li>• Loops: while, for, range(), loops control, and nesting of loops.</li></ul>                                                                                                                                                                                                                                                                                                                                                                          |
| Thema 3 | <ul style="list-style-type: none"><li>• parameters, arguments, and scopes. It also covers Recursion, Exception hierarchy, Exception handling, etc.</li></ul>                                                                                                                                                                                                                                                                                                                                |
| Thema 4 | <ul style="list-style-type: none"><li>• Control Flow: This section covers conditional statements such as if, if-else, if-elif, if-elif-else</li></ul>                                                                                                                                                                                                                                                                                                                                       |
| Thema 5 | <ul style="list-style-type: none"><li>• Data Collections: In this section, the focus is on list construction, indexing, slicing, methods, and comprehensions; it covers Tuples, Dictionaries, and Strings.</li></ul>                                                                                                                                                                                                                                                                        |

## Python Institute PCEP - Certified Entry-Level Python Programmer PCEP-30-02 Prüfungsfragen mit Lösungen (Q13-Q18):

### 13. Frage

What is true about tuples? (Select two answers.)

- A. Tuples can be indexed and sliced like lists.
- B. Tuples are immutable, which means that their contents cannot be changed during their lifetime.
- C. The len { } function cannot be applied to tuples.
- D. An empty tuple is written as { } .

**Antwort: A,B**

Begründung:

Tuples are one of the built-in data types in Python that are used to store collections of data. Tuples have some characteristics that distinguish them from other data types, such as lists, sets, and dictionaries. Some of these characteristics are:

\* Tuples are immutable, which means that their contents cannot be changed during their lifetime. Once a tuple is created, it cannot be modified, added, or removed. This makes tuples more stable and reliable than mutable data types. However, this also means that tuples are less flexible and dynamic than mutable data types. For example, if you want to change an element in a tuple, you have to create a new tuple with the modified element and assign it to the same variable<sup>12</sup>

\* Tuples are ordered, which means that the items in a tuple have a defined order and can be accessed by using their index. The index of a tuple starts from 0 for the first item and goes up to the length of the tuple minus one for the last item. The index can also be negative, in which case it counts from the end of the tuple. For example, if you have a tuple `t = ("a", "b", "c")`, then `t[0]` returns "a", and `t`

`[-1]` returns "c"<sup>12</sup>

\* Tuples can be indexed and sliced like lists, which means that you can get a single item or a sublist of a tuple by using square brackets and specifying the start and end index. For example, if you have a tuple `t = ("a", "b", "c", "d", "e")`, then `t[2]` returns "c", and `t[1:4]` returns ("b", "c", "d"). Slicing does not raise any exception, even if the start or end index is out of range. It will just return an empty tuple or the closest possible sublist<sup>12</sup>

\* Tuples can contain any data type, such as strings, numbers, booleans, lists, sets, dictionaries, or even other tuples. Tuples can also have duplicate values, which means that the same item can appear more than once in a tuple. For example, you can have a tuple `t = (1, 2, 3, 1, 2)`, which contains two 1s and two

2s<sup>12</sup>

\* Tuples are written with round brackets, which means that you have to enclose the items in a tuple with parentheses. For example, you can create a tuple `t = ("a", "b", "c")` by using round brackets. However, you can also create a tuple without using round brackets, by just separating the items with commas. For example, you can create the same tuple `t = "a", "b", "c"` by using commas.

This is called tuple packing, and it allows you to assign multiple values to a single variable<sup>12</sup>

\* The len() function can be applied to tuples, which means that you can get the number of items in a tuple by using the len() function. For example, if you have a tuple t = ("a", "b", "c"), then len(t) returns 3<sup>12</sup>

\* An empty tuple is written as (), which means that you have to use an empty pair of parentheses to create a tuple with no items. For example, you can create an empty tuple t = () by using empty parentheses.

However, if you want to create a tuple with only one item, you have to add a comma after the item, otherwise Python will not recognize it as a tuple. For example, you can create a tuple with one item t = ("a",) by using a comma<sup>12</sup>. Therefore, the correct answers are A. Tuples are immutable, which means that their contents cannot be changed during their lifetime. and D. Tuples can be indexed and sliced like lists.

Reference: Python Tuples - W3Schools Tuples in Python - GeeksforGeeks

#### 14. Frage

Insert the code boxes in the correct positions in order to build a line of code which asks the user for an Integer value and assigns it to the depth variable.

(Note: some code boxes will not be used.)

input (

"Enter immersion depth:" )

)

float (

=

int (

depth



Back to Review

Antwort:

Begründung:

input (

"Enter immersion depth:" )

)

float (

=

int (

depth

in "Enter immersion ) h:"



Back to Review



### 15. Frage

Assuming that the phone\_dir dictionary contains name:number pairs, arrange the code boxes to create a valid line of code which adds Oliver Twist's phone number (5551122333) to the directory.



**Antwort:**

Begründung:

phone\_dir["Oliver Twist"] = ["5551122333"]

Explanation:



To correctly add Oliver Twist's phone number to the phone\_dir dictionary, the code must follow this phone\_dir["Oliver Twist"] = ["5551122333"] Now, let's match that with your code boxes and arrange them:

```
* phone_dir
* [
* "Oliver Twist"
* ]
* =
* [
* "5551122333"
* ]
```

Final Order: phone\_dir [ "Oliver Twist" ] = [ "5551122333" ]

### 16. Frage

What is the expected output of the following code?



- A. 0
- B. The code is erroneous and cannot be run.
- C. 12.849.923.2

- D. yh

**Antwort: C**

### 17. Frage

What is true about exceptions and debugging? (Select two answers.)

- A. The default (anonymous) except branch cannot be the last branch in the try-except block.
- B. If some Python code is executed without errors, this proves that there are no errors in it.
- **C. A tool that allows you to precisely trace program execution is called a debugger.**
- **D. One try-except block may contain more than one except branch.**

**Antwort: C,D**

Begründung:

Exceptions and debugging are two important concepts in Python programming that are related to handling and preventing errors. Exceptions are errors that occur when the code cannot be executed properly, such as syntax errors, type errors, index errors, etc. Debugging is the process of finding and fixing errors in the code, using various tools and techniques. Some of the facts about exceptions and debugging are:

\* A tool that allows you to precisely trace program execution is called a debugger. A debugger is a program that can run another program step by step, inspect the values of variables, set breakpoints, evaluate expressions, etc. A debugger can help you find the source and cause of an error, and test possible solutions. Python has a built-in debugger module called `pdb`, which can be used from the command line or within the code. There are also other third-party debuggers available for Python, such as PyCharm, Visual Studio Code, etc.<sup>12</sup>

\* If some Python code is executed without errors, this does not prove that there are no errors in it. It only means that the code did not encounter any exceptions that would stop the execution. However, the code may still have logical errors, which are errors that cause the code to produce incorrect or unexpected results. For example, if you write a function that is supposed to calculate the area of a circle, but you use the wrong formula, the code may run without errors, but it will give you the wrong answer. Logical errors are harder to detect and debug than syntax or runtime errors, because they do not generate any error messages. You have to test the code with different inputs and outputs, and compare them with the expected results.<sup>34</sup>

\* One try-except block may contain more than one except branch. A try-except block is a way of handling exceptions in Python, by using the keywords `try` and `except`. The `try` block contains the code that may raise an exception, and the `except` block contains the code that will execute if an exception occurs. You can have multiple `except` blocks for different types of exceptions, or for different actions to take. For example, you can write a try-except block like this:

```
try: # some code that may raise an exception
except ValueError: # handle the ValueError exception
except ZeroDivisionError: # handle the ZeroDivisionError exception
except: # handle any other exception
```

This way, you can customize the error handling for different situations, and provide more informative messages or alternative solutions.<sup>5</sup>

\* The default (anonymous) except branch can be the last branch in the try-except block. The default except branch is the one that does not specify any exception type, and it will catch any exception that is not handled by the previous except branches. The default except branch can be the last branch in the try- except block, but it cannot be the first or the only branch. For example, you can write a try-except block like this:

```
try: # some code that may raise an exception
except ValueError: # handle the ValueError exception
except: # handle any other exception
```

This is a valid try-except block, and the default except branch will be the last branch. However, you cannot write a try-except block like this:

```
try: # some code that may raise an exception
except: # handle any exception
```

This is an invalid try-except block, because the default except branch is the only branch, and it will catch all exceptions, even those that are not errors, such as `KeyboardInterrupt` or `SystemExit`. This is considered a bad practice, because it may hide or ignore important exceptions that should be handled differently or propagated further. Therefore, you should always specify the exception types that you want to handle, and use the default except branch only as a last resort.<sup>5</sup> Therefore, the correct answers are A. A tool that allows you to precisely trace program execution is called a debugger. and C. One try-except block may contain more than one except branch.

Reference: Python Debugger - Python `pdb` - [GeeksforGeeks](#)How can I see the details of an exception in Python's debugger?Python Debugging (fixing problems)Python - start interactive debugger when exception would be otherwise thrownPython Try Except [Error Handling and Debugging - Programming with Python for Engineers]

### 18. Frage

.....

100% Garantie für PCEP-30-02 Zertifizierung PCEP - Certified Entry-Level Python Programmer Prüfungserfolg. Wenn Sie ZertFragen PCEP-30-02 Prüfung Python Institute wählen, ist ZertFragen Test Engine das perfekte Werkzeug, mit dem Sie sich

besser auf die Zertifizierungsprüfung vorbereiten. Erfolg kommt einfach, wenn Sie mit Hilfe PCEP-30-02 Dumps (PCEP - Certified Entry-Level Python Programmer) von ZertFragen nutzen. Falls Sie in der Prüfung durchfallen, geben wir Ihnen eine volle Rückerstattung Ihres Einkaufs.

**PCEP-30-02 Lernhilfe:** [https://www.zertfragen.com/PCEP-30-02\\_pruefung.html](https://www.zertfragen.com/PCEP-30-02_pruefung.html)

- Kostenlose gültige Prüfung Python Institute PCEP-30-02 Sammlung - Examcollection □ Geben Sie ► [www.echtefrage.top](http://www.echtefrage.top) □ ein und suchen Sie nach kostenloser Download von ► PCEP-30-02 □ □ PCEP-30-02 Fragenkatalog
- PCEP-30-02 Prüfungsfragen, PCEP-30-02 Fragen und Antworten, PCEP - Certified Entry-Level Python Programmer □ Öffnen Sie die Website □ [www.itzert.com](http://www.itzert.com) □ Suchen Sie □ PCEP-30-02 □ Kostenloser Download □ PCEP-30-02 PDF Testsoftware
- PCEP-30-02 Online Prüfung □ PCEP-30-02 Prüfungssimulationen □ PCEP-30-02 PDF Testsoftware □ Suchen Sie jetzt auf ► [www.deutschpruefung.com](http://www.deutschpruefung.com) ◄ nach ✓ PCEP-30-02 □ ✓ □ und laden Sie es kostenlos herunter □ PCEP-30-02 Testfragen
- PCEP-30-02 Fragen - Antworten - PCEP-30-02 Studienführer - PCEP-30-02 Prüfungsvorbereitung □ Suchen Sie auf der Webseite □ [www.itzert.com](http://www.itzert.com) □ nach ► PCEP-30-02 □ und laden Sie es kostenlos herunter ◉ PCEP-30-02 Online Prüfung
- PCEP-30-02 Zertifikatsdemo □ PCEP-30-02 Vorbereitung □ PCEP-30-02 Online Prüfung □ Suchen Sie auf □ [www.zertpruefung.de](http://www.zertpruefung.de) □ nach ► PCEP-30-02 □ und erhalten Sie den kostenlosen Download mühelos □ PCEP-30-02 Deutsche Prüfungsfragen
- PCEP-30-02 Online Prüfungen □ PCEP-30-02 Testantworten □ PCEP-30-02 Zertifikatsdemo □ Sie müssen nur zu **【 [www.itzert.com](http://www.itzert.com) 】** gehen um nach kostenloser Download von ► PCEP-30-02 □ zu suchen □ PCEP-30-02 Deutsch Prüfungsfragen
- PCEP-30-02 PDF Testsoftware □ PCEP-30-02 Vorbereitung □ PCEP-30-02 Prüfungsfragen □ Öffnen Sie ► [www.zertpruefung.de](http://www.zertpruefung.de) ◄ geben Sie ► PCEP-30-02 □ ein und erhalten Sie den kostenlosen Download □ PCEP-30-02 Deutsche
- PCEP-30-02 Prüfungsfragen, PCEP-30-02 Fragen und Antworten, PCEP - Certified Entry-Level Python Programmer □ Suchen Sie jetzt auf 「 [www.itzert.com](http://www.itzert.com) 」 nach ► PCEP-30-02 □ und laden Sie es kostenlos herunter □ PCEP-30-02 Kostenlos Downladen
- PCEP-30-02 Prüfungsfragen, PCEP-30-02 Fragen und Antworten, PCEP - Certified Entry-Level Python Programmer ♥ □ URL kopieren ► [www.echtefrage.top](http://www.echtefrage.top) □ Öffnen und suchen Sie ( PCEP-30-02 ) Kostenloser Download □ PCEP-30-02 Zertifizierung
- Kostenlose gültige Prüfung Python Institute PCEP-30-02 Sammlung - Examcollection i Suchen Sie jetzt auf “ [www.itzert.com](http://www.itzert.com) ” nach ( PCEP-30-02 ) um den kostenlosen Download zu erhalten □ PCEP-30-02 Prüfungssimulationen
- PCEP-30-02 Testantworten □ PCEP-30-02 Testantworten □ PCEP-30-02 Online Prüfung □ Suchen Sie auf der Webseite ► [www.zertsoft.com](http://www.zertsoft.com) □ nach ► PCEP-30-02 □ und laden Sie es kostenlos herunter ☒ PCEP-30-02 Deutsch Prüfungsfragen
- [daotao.wisebusiness.edu.vn](http://daotao.wisebusiness.edu.vn), [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [attamhidfoundation.com](http://attamhidfoundation.com), [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [shortcourses.russellcollege.edu.au](http://shortcourses.russellcollege.edu.au), [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [lms.ait.edu.za](http://lms.ait.edu.za), Disposable vapes

BONUS!!! Laden Sie die vollständige Version der ZertFragen PCEP-30-02 Prüfungsfragen kostenlos herunter:  
<https://drive.google.com/open?id=1-heru-TM77fTSUOJ89PsX1UjLThhaiK8>